

ARITHMETIC PROCESSOR

Patent Number: JP11167488
Publication date: 1999-06-22
Inventor(s): SHINOHARA MAKOTO
Applicant(s):: TOSHIBA CORP
Requested Patent:  JP11167488
Application Number: JP19970334369 19971204
Priority Number(s):
IPC Classification: G06F9/30 ; G06F9/30 ; G06F9/45
EC Classification:
Equivalents:

Abstract

PROBLEM TO BE SOLVED: To improve object code generation efficiency at an RISC microprocessor called a reduced instruction set computer.

SOLUTION: An redundant part of a 32-bit reduced instruction set consisting, for example, of op2h and op21 is compressed to generate a 16-bit compressed instruction set op2. Then, after housing this set op2 within a memory 11, the above is read by a predecoder 13 at the time of executing processing to covert to the 32-bit reduced instruction sets op2h and op21, which are fetched by an IR 14. An object code size is reduced like this to suppress the increase in the capacity of the memory 11.

Data supplied from the **esp@cenet** database - I2

THIS PAGE BLANK (USPTO)

(19)日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11)特許出願公開番号

特開平11-167488

(43)公開日 平成11年(1999)6月22日

(51)Int.Cl. ⁶	識別記号	F I
G 0 6 F 9/30	3 1 0	G 0 6 F 9/30
	3 5 0	3 1 0 A
		3 5 0 G
		3 5 0 E
9/45		9/44 3 2 2 E

審査請求 未請求 請求項の数11 O L (全 6 頁)

(21)出願番号 特願平9-334369

(22)出願日 平成9年(1997)12月4日

(71)出願人 000003078

株式会社東芝

神奈川県川崎市幸区堀川町72番地

(72)発明者 篠原 誠

神奈川県川崎市幸区堀川町580番1号 株

式会社東芝半導体システム技術センター内

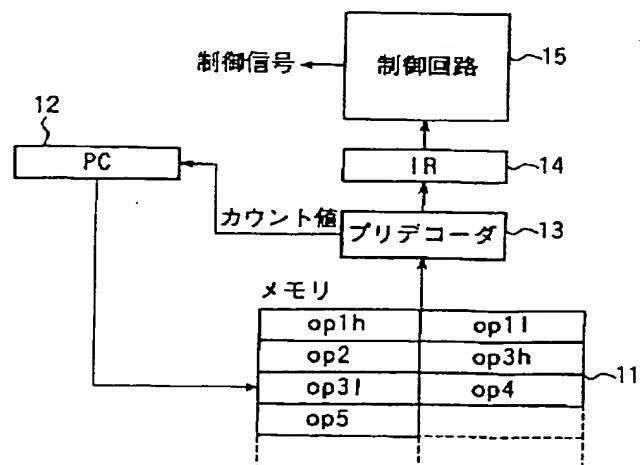
(74)代理人 弁理士 鈴江 武彦 (外6名)

(54)【発明の名称】 演算処理装置

(57)【要約】

【課題】本発明は、縮小命令セットコンピュータと称されるRISC型のマイクロプロセッサにおいて、オブジェクトコード効率を向上できるようにすることを最も主要な特徴とする。

【解決手段】たとえば、op2h, op2lからなる32ビットの縮小命令セットの、その冗長な部分を圧縮して、16ビットの圧縮命令セットop2を生成する。そして、この圧縮命令セットop2をメモリ11内に格納しておき、処理を実行する際にプリデコーダ13により読み出して、32ビットの縮小命令セットop2h, op2lに変換した後、IR14にフェッチさせる。こうして、オブジェクトコードサイズを小さくし、メモリ11の容量が増大するのを抑える構成となっている。



【特許請求の範囲】

【請求項1】 固定長の命令の冗長な部分を、あらかじめ決められた規則にしたがって圧縮した圧縮命令を格納する格納部と、

この格納部内に格納された前記圧縮命令を読み出して、その冗長な部分を伸張して元の固定長の命令に変換する変換部と、

この変換部を介して供給される固定長の命令にもとづいて演算処理を実行する処理部とを具備したことを特徴とする演算処理装置。

【請求項2】 前記格納部は、前記圧縮命令および固定長の命令を一括して格納することを特徴とする請求項1に記載の演算処理装置。

【請求項3】 前記格納部内に格納された、前記圧縮命令または固定長の命令の格納位置を指示する指示部をさらに備えることを特徴とする請求項1または請求項2のいずれかに記載の演算処理装置。

【請求項4】 前記変換部は、前記格納部内に格納された固定長の命令を伸張せずに前記処理部に供給する手段を有することを特徴とする請求項1ないし請求項3のいずれかに記載の演算処理装置。

【請求項5】 前記変換部は、前記格納部内より読み出した命令が圧縮命令か固定長の命令かに応じて、前記指示部により指示される次の命令の格納位置を制御する手段を有することを特徴とする請求項1ないし請求項4のいずれかに記載の演算処理装置。

【請求項6】 少なくとも、前記格納部、前記変換部、および、前記処理部は、1チップ化されていることを特徴とする請求項1に記載の演算処理装置。

【請求項7】 命令長が固定長の縮小命令、および、固定長の縮小命令の冗長な部分を、あらかじめ決められた規則にしたがって圧縮した圧縮命令を一括して格納する格納部と、

この格納部内に格納された前記縮小命令または前記圧縮命令を読み出し、読み出した前記圧縮命令の、その冗長な部分を伸張して元の固定長の縮小命令に変換する変換部と、

この変換部を介して供給される固定長の縮小命令にもとづいて演算処理を実行する処理部とを具備したことを特徴とする演算処理装置。

【請求項8】 前記格納部内に格納された、前記圧縮命令または固定長の縮小命令の格納位置を指示する指示部をさらに備えることを特徴とする請求項7に記載の演算処理装置。

【請求項9】 前記変換部は、前記格納部内に格納された固定長の縮小命令を伸張せずに前記処理部に供給する手段を有することを特徴とする請求項7に記載の演算処理装置。

【請求項10】 前記変換部は、前記格納部内より読み出した命令が圧縮命令か固定長の縮小命令かに応じて、

前記指示部により指示される次の命令の格納位置を制御する手段を有することを特徴とする請求項7に記載の演算処理装置。

【請求項11】 少なくとも、前記格納部、前記変換部、および、前記処理部は、1チップ化されていることを特徴とする請求項7に記載の演算処理装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】この発明は、演算処理装置にかかり、特に、縮小命令セットコンピュータと称されるRISC (Reduced Instruction Set Computers) 型のマイクロプロセッサ (マイクロコンピュータともいう) に関するものである。

【0002】

【従来の技術】近年、自動車や家電製品においては、コンピュータによる制御を可能とするために、1チップ化されたマイクロプロセッサが組み込まれている。一般に、自動車や家電製品に組み込まれるマイクロプロセッサは、載置スペースの制限などから、システムの小型化が求められている。また、システムに登載されるプログラムのオブジェクトコードサイズも小さくしなければならない。

【0003】そのため、従来は、8ビット〜64ビット程度の複雑な命令セットを用いて処理を行う、いわゆるCISC (Complex Instruction Set Computer) 型のマイクロプロセッサが主流であった。

【0004】しかしながら、システムが大規模化し、また、複雑化してきている昨今、従来のCISC型のマイクロプロセッサでは対応しきれなくなりつつある。それは、CISC型のマイクロプロセッサでの処理の高速化に限界が見え始めたためである。

【0005】そこで、組み込み型のマイクロプロセッサとして、RISC型のマイクロプロセッサが注目されている。図2は、従来のRISC型のマイクロプロセッサの基本的構成を示すものである。

【0006】このシステムは、固定長 (命令長が一定長) の複数の命令セットを格納するメモリ1、このメモリ1内における所望の命令セットのアドレスポイントを指示するプログラムカウンタ (PC) 2、このPC2からのアドレスポイントに対応して、上記メモリ1内より読み出された命令セットをフェッチする命令レジスタ

(IR) 3、および、このIR3によってフェッチされた命令セットにしたがって各種の処理を実行する制御回路4を有して構成されている。

【0007】このシステムの場合、通常は、たとえば、32ビットに統一された単純な縮小命令セットを用いるため、CISC型のマイクロプロセッサに比べ、システムの簡素化および処理の高速化が可能である。

【0008】しかしながら、RISC型のマイクロプロセッサは、命令セット (たとえば、第1の縮小命令セッ

top1h, op1l、第2の縮小命令セットop2h, op2l、第3の縮小命令セットop3h, op3l、第4の縮小命令セットop4h, op4l、第5の縮小命令セットop5h, op5l)がいずれも固定長のため、同一のソースプログラムから生成したオブジェクトコードサイズが、CISC型のマイクロプロセッサよりも大きくなるという不具合があった。これは、すべての命令セットを固定長とするために、少なくとも命令セットの幾つかに冗長な部分が多く含まれるためである。

【0009】

【発明が解決しようとする課題】上記したように、従来においては、システムの大規模化、複雑化に対しては、RISC型のマイクロプロセッサにより容易に対応できるものの、RISC型のマイクロプロセッサの場合、プログラムのオブジェクトコードサイズがCISC型のマイクロプロセッサよりも大きくなるため、組み込み型のマイクロプロセッサとしては適さないという不具合があった。

【0010】そこで、この発明は、オブジェクトコード効率を向上でき、組み込み型のマイクロプロセッサとして用いて好適な演算処理装置を提供することを目的としている。

【0011】

【課題を解決するための手段】上記の目的を達成するために、この発明の演算処理装置にあっては、固定長の命令の冗長な部分を、あらかじめ決められた規則にしたがって圧縮した圧縮命令を格納する格納部と、この格納部に格納された前記圧縮命令を読み出して、その冗長な部分を伸張して元の固定長の命令に変換する変換部と、この変換部を介して供給される固定長の命令にもとづいて演算処理を実行する処理部とから構成されている。

【0012】また、この発明の演算処理装置にあっては、命令長が固定長の縮小命令、および、固定長の縮小命令の冗長な部分を、あらかじめ決められた規則にしたがって圧縮した圧縮命令を一括して格納する格納部と、この格納部に格納された前記縮小命令または前記圧縮命令を読み出し、読み出した前記圧縮命令の、その冗長な部分を伸張して元の固定長の縮小命令に変換する変換部と、この変換部を介して供給される固定長の縮小命令にもとづいて演算処理を実行する処理部とから構成されている。

【0013】この発明の演算処理装置によれば、マイクロプロセッサのオブジェクトコードサイズを小さくできるようになる。これにより、RISC型のマイクロプロセッサの特長を生かした高速な組み込み型のマイクロプロセッサを、そのオブジェクトを含む命令を格納する格納部の容量をCISC型のマイクロプロセッサに比べて大幅に増やすことなく、容易に実現することが可能となるものである。

【0014】特に、格納部内に一括して格納される既存の固定長の命令をそのまま実行できるようにした場合には、既にあるソフトウェア資産を活かすことが可能となるものである。

【0015】

【発明の実施の形態】以下、この発明の実施の形態について図面を参照して説明する。図1は、本発明の実施の一形態にかかるRISC型のマイクロプロセッサの、基本構成を概略的に示すものである。なお、ここでは、組み込み型のマイクロプロセッサとして、32ビットRISC型1チップマイコン（マイクロコンピュータ）に適用した場合の例を説明する。

【0016】このシステムは、たとえば、複数の命令セットを一括して格納する格納部としてのメモリ11、このメモリ11内における所望の命令セットのアドレスポイント（格納位置）を指示する指示部としてのプログラムカウンタ（以下、単にPCと略記する）12、このPC12からのアドレスポイントに対応する、上記メモリ11内に格納された命令セットを読み出すプリデコーダ（変換部）13、このプリデコーダ13によって、上記メモリ11内より読み出された命令セットがフェッチされる命令レジスタ（以下、単にIRと略記する）14、および、このIR14によってフェッチされた命令セットにしたがって各種の処理を実行する制御回路15が、1チップ化されてなる構成とされている。

【0017】上記メモリ11には、たとえば、既存の32ビット（固定長）の命令セットop1h, op1l（第1の縮小命令セット）、および、命令セットop3h, op3l（第3の縮小命令セット）が、それぞれ、所定のアドレスポイントに格納されている。

【0018】また、メモリ11内には、既存の32ビットの第2の縮小命令セットop2h, op2l、第4の縮小命令セットop4h, op4l、および、第5の縮小命令セットop5h, op5lの、その冗長な部分をあらかじめ決められた規則にしたがって圧縮した、たとえば16ビットの圧縮命令セットop2、圧縮命令セットop4、および、圧縮命令セットop5が、それぞれ、所定のアドレスポイントに格納されるようになっている。

【0019】なお、32ビットの縮小命令セットの固定長の冗長な部分を、あらかじめ決められた規則にしたがって圧縮し、16ビットの圧縮命令セットを生成する方法については後述する。

【0020】上記PC12は、1つの命令セットに応じた処理を実行するごとに、上記プリデコーダ13からのカウント値にしたがって、逐次、次の命令セットのアドレスポイントを示すように、現在のアドレスポイントをカウントアップする機能を有している。

【0021】上記プリデコーダ13は、たとえば、上記メモリ11から読み出した命令セットが、縮小命令セッ

トか圧縮命令セットかをオペレーション (op) コード (以下、単にオペコードと略記する) により判別し、縮小命令セットの場合にはそのまま、圧縮命令セットの場合には、それを伸張して元の縮小命令セットに変換した後に、それぞれ、上記 I R 1 4 に供給するものである。

【0022】すなわち、上記プリデコーダ13は、上記メモリ11から第1の縮小命令セット op 1 h, op 1 l、または、第3の縮小命令セット op 3 h, op 3 l のいずれかを読み出した場合には、32ビットのまま上記 I R 1 4 にフェッチさせるようになっている。

【0023】また、プリデコーダ13は、上記メモリ11から圧縮命令セット op 2、圧縮命令セット op 4、または、圧縮命令セット op 5 のいずれかを読み出した場合には、それぞれ、16ビットに圧縮されている冗長な部分を伸張して元の32ビットの第2の縮小命令セット op 2 h, op 2 l、第4の縮小命令セット op 4 h, op 4 l、および、第5の縮小命令セット op 5 h, op 5 l に変換した後に、上記 I R 1 4 にフェッチさせるようになっている。

【0024】プリデコーダ13としては、たとえば、上記メモリ11内に格納されている全命令セットのオペコードと、全オペコードのうち、少なくとも圧縮命令セットの各オペコードが示す、16ビットの圧縮命令セットを32ビットの縮小命令セットに変換 (伸張) するための規則とを、それぞれに対応させて記憶するテーブルなどを有している。

【0025】この場合、たとえば、上記メモリ11内に格納されている全命令セットのうち、圧縮命令セットの各オペコードと、各オペコードが示す、16ビットの圧縮命令セットを32ビットの縮小命令セットに変換するための規則とを、それぞれに対応させて記憶させておき、読み出した命令セットのオペコードが記憶されていない場合、その命令セットは変換する必要がない32ビットの縮小命令セットであると判別するようにしても良い。

【0026】また、上記プリデコーダ13は、上記メモリ11内より読み出した命令セットが、縮小命令セットか圧縮命令セットかを判別した結果にもとづいて、上記 P C 1 2 に対する、アドレスポイントをカウントアップさせるためのカウント値を制御する機能を有している。

【0027】たとえば、上記メモリ11内より読み出した命令セットが縮小命令セットの場合には、アドレスポイントが32ビット (4バイト) 分だけカウントアップするように、カウント値を制御するようになっている。

【0028】一方、圧縮命令セットの場合には、たとえば、アドレスポイントが16ビット (2バイト) 分だけカウントアップするように、カウント値を制御するようになっている。

【0029】上記 I R 1 4 は、上記プリデコーダ13より供給される命令セットを、常に、32ビットの縮小命

令セットの形でフェッチするものである。ここで、32ビットの縮小命令セットから、16ビットの圧縮命令セットを生成するための変換 (圧縮) の方法について説明する。圧縮命令セットは、たとえば下記の表1に示すように、通常のアセンブラで16ビットに圧縮できるものを、自動的に圧縮したコードにより生成される。

【0030】

【表1】

32→16ビット変換例

(a) LOAD命令			
32ビット長の命令セット			
LB rt, offset(base)			
LB	base	rt	offset
100000
<-6->	<-5->	<-5->	<-16->
<-----32----->			
16ビット長の命令セット			
LB rt, offset(0)			
LB	base	rt	
110000	
<-6->	<-5->	<-5->	
<-----16----->			
(b) ADD命令			
32ビット長の命令セット			
ADDIU rt, ra, immediate			
ADDIU	ra	rt	immediate
001001
<-6->	<-5->	<-5->	<-16->
<-----32----->			
16ビット長の命令セット			
ADDIU rt, rt, imm6			
ADDIU	rt	immediate5	
110101	
<-6->	<-5->	<-5->	
<-----16----->			
(c) ANDI命令			
32ビット長の命令セット			
ANDI rt, ra, immediate			
ANDI	ra	rt	immediate
001100
<-6->	<-5->	<-5->	<-16->
<-----32----->			
16ビット長の命令セット			
ANDI rt, rt, IPPPP			
ANDI	rt	IPPPPP	
110100	
<-6->	<-5->	<-5->	
<-----16----->			

【0031】たとえば、32ビットの縮小命令セットが、6ビットのオペコードLB (100000)、5ビットのbase (ベースレジスタ)、5ビットのrt (ターゲットレジスタ (ロード先レジスタ))、および、16ビットのoffset (オフセット値) からなるLOAD命令の場合、この命令は、「base+offset」を実行アドレスとして、その実行アドレスのメモリ内容を「rt」内に読み取ることを意味するものであるため、オフセット値が「0」であることが多い。

【0032】そこで、offsetの部分 (16ビット) を削除することで、32ビットのLOAD命令 (縮小命令セット) から、6ビットのLB (11000

0)、5ビットのbase、および、5ビットのrtからなる、16ビットのLOAD命令(圧縮命令セット)が生成される。

【0033】また、たとえば、32ビットの縮小命令セットが、6ビットのオペコードADDIU(001001)、5ビットのrs、5ビットのrt、および、16ビットのimmediateからなるADD命令の場合、この命令は、「rs+immediate」の算術演算結果を「rt」に格納することを意味するものであって、「rs=rt」の場合、immediate値が4ビット以下(5ビットで十分なほど小さいとき)であることが多い。

【0034】そこで、rsの部分(5ビット)とimmediateの一部分(11ビット)を削除することで、32ビットのADD命令(縮小命令セット)から、6ビットのADDIU(110T01)、5ビットのrt(rt, rt)、および、5ビットのimm5(immediate5)からなる、16ビットのADD命令(圧縮命令セット)が生成される。

【0035】さらに、たとえば、32ビットの縮小命令セットが、6ビットのオペコードANDI(001100)、5ビットのrs、5ビットのrt、および、16ビットのimmediateからなるANDI命令の場合、この命令は、「rs&immediate」の論理演算結果を「rt」に格納することを意味するものであって、「rs=rt」の場合、immediate値が特定の1ビットのみ1であることが多い。

【0036】そこで、そのビット位置を示す4ビット「PPPP」と特定のビットが1か0かを示す1ビットとでrsの部分(5ビット)を置き換えるとともに、immediateの部分(16ビット)を削除することで、32ビットのANDI命令(縮小命令セット)から、6ビットのANDI(110100)、5ビットのrt(rt, rt)、および、5ビットの1PPPPからなる、16ビットのANDI命令(圧縮命令セット)が生成される。

【0037】同様にして、32ビットの縮小命令セットの、その冗長な部分をそれぞれに圧縮することにより、上記したLOAD命令、ADD命令、ANDI命令以外にも、各種の16ビットの圧縮命令セットが生成できる。

【0038】たとえば、32ビットのLOAD命令を第2の縮小命令セットop2h, op2lとした場合、これを変換した16ビットのLOAD命令を圧縮命令セットop2として、また、32ビットのADD命令を第4の縮小命令セットop4h, op4lとした場合、これを変換した16ビットのADD命令を圧縮命令セットop4として、さらに、32ビットのANDI命令を第5の縮小命令セットop5h, op5lとした場合、これを変換した16ビットのANDI命令を圧縮命令セット

op5として、それぞれ、メモリ11内に格納することにより、従来のRISC型のマイクロプロセッサ(図2参照)に比べ、図示破線の方だけ、プログラムのオブジェクトコードサイズを小さくすることが可能となる。

【0039】この結果、RISC型のマイクロプロセッサの特長である高速性を損うことなく、また、メモリ容量をCISC型のマイクロプロセッサに比べて大幅に増やすことなく、組み込み型のマイクロプロセッサを容易に実現できる。

【0040】上記したように、RISC型のマイクロプロセッサのオブジェクトコードサイズを小さくできるようにしている。すなわち、32ビットの縮小命令セットの、その冗長な部分を圧縮して命令長を変化させることにより、RISC型のマイクロプロセッサのオブジェクトコードサイズを改善するようにしている。

【0041】これにより、RISC型のマイクロプロセッサの特長である高速性を損うことなく、しかも、プリデコードなる付加回路を追加するのみで、RISC型のマイクロプロセッサの構成を大幅に変更することなく、オブジェクトコード効率の良いRISC型のマイクロプロセッサを構成できるようになる。

【0042】したがって、メモリ容量をCISC型のマイクロプロセッサに比べて大幅に増やしたりすることなしに、RISC型のマイクロプロセッサによって、組み込み型のマイクロプロセッサを容易に実現できるものである。

【0043】しかも、既存の32ビットの縮小命令セットについては、プリデコードを通過させて、そのまま実行できるようにしている。このため、既にあるソフトウェア資産を十二分に活かすことが可能である。

【0044】なお、上記した本発明の実施の一形態においては、RISC型のマイクロプロセッサに適用した場合を例に説明したが、これに限らず、たとえばDSP(Digital Signal Processor)型のマイクロプロセッサにも同様に適用できる。

【0045】また、生成する圧縮命令セットやその方法については、マイクロプロセッサの種類や命令セットの発生の頻度などに応じて、適宜、効率的な手段を選択するようにすれば良い。その他、この発明の要旨を変えない範囲において、種々変形実施可能なことは勿論である。

【0046】

【発明の効果】以上、詳述したようにこの発明によれば、オブジェクトコード効率を向上でき、組み込み型のマイクロプロセッサとして用いて好適な演算処理装置を提供できる。

【図面の簡単な説明】

【図1】この発明の実施の一形態にかかる、RISC型のマイクロプロセッサを示す概略構成図。

【図2】従来技術とその問題点を説明するために示す、

RISC型のマイクロプロセッサの概路構成図。

【符号の説明】

- 11…メモリ
- 12…プログラムカウンタ (PC)
- 13…プリデコーダ
- 14…命令レジスタ (IR)
- 15…制御回路

op1h, op1l…第1の縮小命令セット (32ビット)

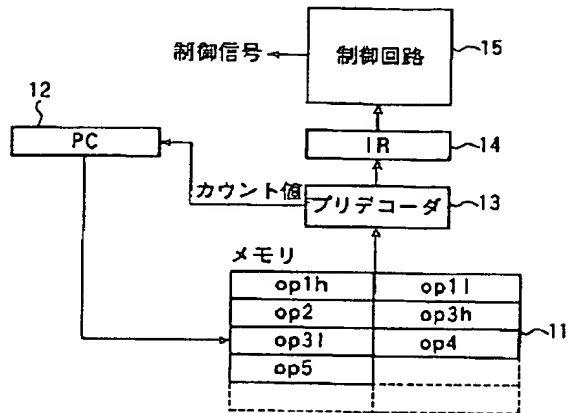
op2…圧縮命令セット (16ビット)

op3h, op3l…第3の縮小命令セット (32ビット)

op4…圧縮命令セット (16ビット)

op5…圧縮命令セット (16ビット)

【図1】



【図2】

